Problem Set 1 Solution

AA274: Principles of Robotic Autonomy — **Chi Zhang**

Stanford University — SUNet ID: czhang94

Winter 2018 — Date: January 24, 2018

Problem 1: Optimal Control

(i) Given the state vector $\mathbf{x} = (x, y, \theta)$ and control vector $\mathbf{u} = (V, \omega)$, it is simple to find the co-state vector $\mathbf{p} \in \mathbb{R}^3$. The Hamiltonian is derived as

$$H(\mathbf{x}, \mathbf{u}, \mathbf{p}) = \lambda + V^2 + \omega^2 + p_1 V \cos(\theta) + p_2 V \sin(\theta) + p_3 \omega$$

The conditions for optimality are

$$\dot{\mathbf{x}}^* = \frac{\partial H}{\partial \mathbf{p}}(\mathbf{x}^*, \mathbf{u}^*, \mathbf{p}^*) = \begin{bmatrix} V^* \cos(\theta^*) \\ V^* \sin(\theta^*) \\ \omega^* \end{bmatrix}$$

$$\dot{\mathbf{p}}^* = -\frac{\partial H}{\partial \mathbf{x}}(\mathbf{x}^*, \mathbf{u}^*, \mathbf{p}^*) = \begin{bmatrix} 0 \\ 0 \\ V^* p_1^* \sin(\theta^*) - V^* p_2^* \cos(\theta^*) \end{bmatrix}$$

$$\mathbf{0} = -\frac{\partial H}{\partial \mathbf{u}}(\mathbf{x}^*, \mathbf{u}^*, \mathbf{p}^*) = \begin{bmatrix} 2V^* + p_1^* \cos(\theta^*) + p_2^* \sin(\theta^*) \\ 2\omega^* + p_3^* \end{bmatrix}$$

Considering control constraints, we also have

$$H(\mathbf{x}^*, \mathbf{u}^*, \mathbf{p}^*) \leq H(\mathbf{x}^*, \mathbf{u}, \mathbf{p}^*)$$

and boundary conditions (BCs) are

$$\mathbf{x}(0) = \begin{bmatrix} 0 \\ 0 \\ -\frac{\pi}{2} \end{bmatrix}, \quad \mathbf{x}(t_f) = \begin{bmatrix} 5 \\ 5 \\ -\frac{\pi}{2} \end{bmatrix}$$

From the above equations, we know

$$V^* = -\frac{p_1^* \cos(\theta^*) + p_2^* \sin(\theta^*)}{2}, \quad \omega^* = -\frac{p_3^*}{2}$$

Define a new state vector $\mathbf{z} = (\mathbf{x}, \mathbf{p}, r)$ where $r = t/t_f \in [0,\ 1]$, then BVP becomes

$$\frac{d\dot{\mathbf{z}}_i^*(\tau)}{d\tau} = \frac{d\dot{\mathbf{x}}^*(\tau)}{d\tau} = t_f \dot{\mathbf{x}}^* = r^*(\tau) \begin{bmatrix} V^*(\tau) \cos(\theta^*(\tau)) \\ V^*(\tau) \sin(\theta^*(\tau)) \\ \omega^*(\tau) \end{bmatrix} \quad (i = 1, 2, 3)$$

$$\frac{d\dot{\mathbf{z}}_i^*(\tau)}{d\tau} = \frac{d\dot{\mathbf{p}}^*(\tau)}{d\tau} = t_f\dot{\mathbf{p}}^* = r^*(\tau)\begin{bmatrix} 0 \\ 0 \\ V^*(\tau)p_1^*(\tau)\sin(\theta^*(\tau)) - V^*(\tau)p_2^*(\tau)\cos(\theta^*(\tau)) \end{bmatrix} \; (i = 4, 5, 6)$$

$$\frac{d\dot{\mathbf{z}}_7^*(\tau)}{d\tau} = \frac{dr}{d\tau} = 0$$

and BCs become

$$\mathbf{x}^*(0) = \begin{bmatrix} 0 \\ 0 \\ -\frac{\pi}{2} \end{bmatrix}, \quad \mathbf{x}^*(1) = \begin{bmatrix} 5 \\ 5 \\ -\frac{\pi}{2} \end{bmatrix}$$

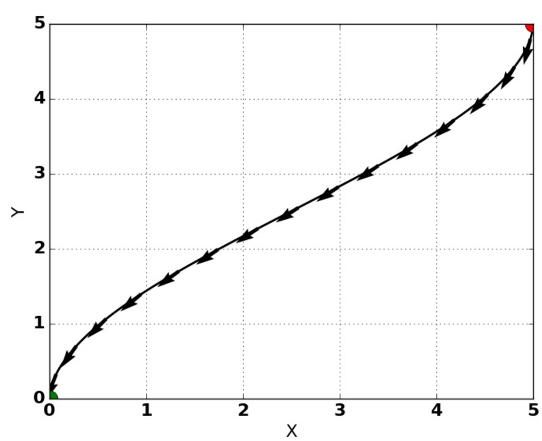
since the final time t_f is free, *i.e.* δt_f is arbitrary, thus

$$H(\mathbf{x}^*(1), \mathbf{u}^*(1), \mathbf{p}^*(1)) = 0$$

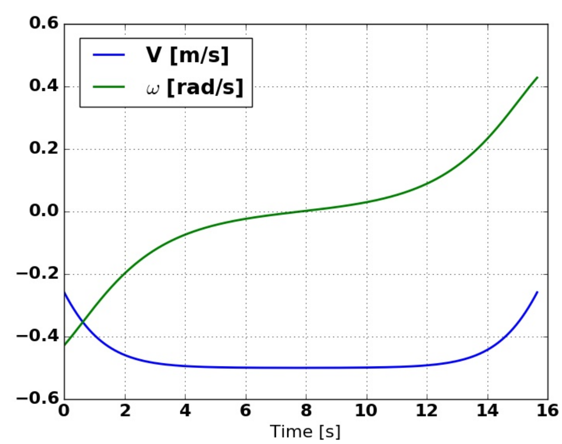
Here, we have seven ODEs for seven unknowns $z_i (i = 1, 2, \cdots, 7)$, which is a 2P-BVP problem.

(iii) In the cost function, λ can be considered as penalty for t_f in optimization. Therefore, the larger λ, the smaller t_f. Since the goal of optimization is to minimize J, using the largest feasible λ yields the smallest t_f. To put it more explicitly, the largest feasible λ will "make" the car move as quick as possible for the smallest t_f, pushing V to limit.

(iv) With $\lambda = 0.25$ and the initial guess $\mathbf{p}_0 = [1.0,\ 1.0,\ -\frac{\pi}{2},\ -1.0,\ -1.0,\ 5.0,\ 10.0]^T$, it can be obtained the figures below:



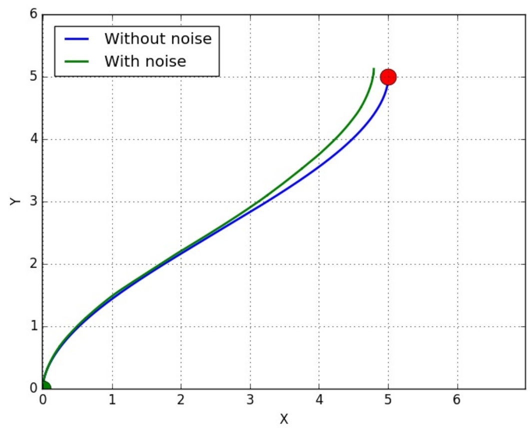
(a) optimal trajectory



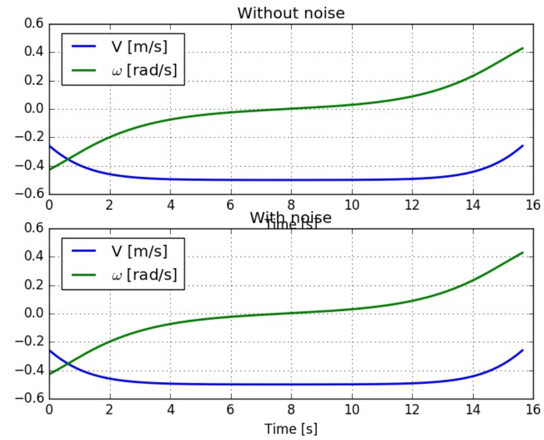
(b) optimal control history

It is trivial to verify the trajectory satisfies BCs and the control history satisfies given constraints, which are $|V| \leq 0.5$m/s and $|\omega| \leq 1.0$rad/s.

(v) The two plots are shown below:



(a) simulation trajectory



(b) simulation control history

Problem 2: Differential Flatness

(i) Given four basis functions $\psi_i (i = 1, 2, 3, 4)$, the initial and final conditions can be rearranged as

$$\begin{bmatrix} x(0) \\ y(0) \\ \dot{x}(0) \\ \dot{y}(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -0.5 \end{bmatrix} \quad \begin{bmatrix} x(t_f) \\ y(t_f) \\ \dot{x}(t_f) \\ \dot{y}(t_f) \end{bmatrix} = \begin{bmatrix} 5 \\ 5 \\ 0 \\ -0.5 \end{bmatrix}$$

Then these conditions can be written in the form of matrix multiplication:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2t_f & 3t_f^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & t_f & t_f^2 & t_f^3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 5 \\ 0 \\ 0 \\ -0.5 \\ 5 \\ -0.5 \end{bmatrix}$$

It can be observed that ω will become undefined if $J = 0$. Also, $\det(J) = V$. If $V(t_f) = 0$, then J becomes non-invertible, leading to singularity issues.

(ii) Solving the linear system above gives us $x, y, \dot{x}, \dot{y}, \ddot{x}, \ddot{y}$, and we already know that

$$\theta = \tan^{-1}(\frac{\dot{y}}{\dot{x}})$$

$$V = \frac{\dot{x}}{\cos\theta} \quad \text{or} \quad V = \frac{\dot{y}}{\sin\theta} \quad \text{or} \quad V = \sqrt{\dot{x}^2 + \dot{y}^2}$$

Also, it can be obtained from kinetic constraints that

$$\ddot{x} = a\cos\theta - \omega(V\sin\theta) = a\cos\theta - \omega\dot{y} \tag{1}$$

$$\ddot{y} = a\sin\theta + \omega(V\cos\theta) = a\sin\theta + \omega\dot{x} \tag{2}$$

Do (2) $\times V\cos\theta -$ (1) $\times V\sin\theta$, we have

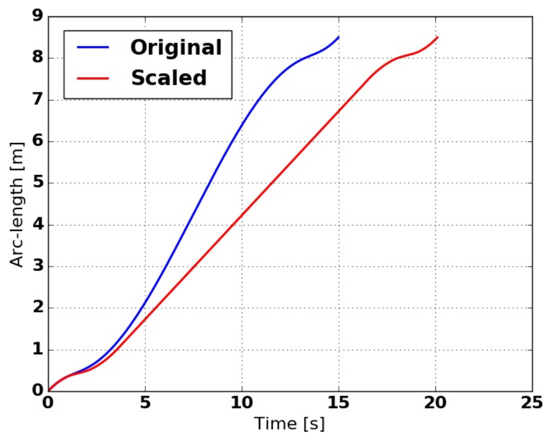
$$\ddot{y}(V\cos\theta) - \ddot{x}(V\sin\theta) = \omega[\dot{x}(V\cos\theta) + \dot{y}(V\sin\theta)]$$

$$\dot{x}\ddot{y} - \dot{y}\ddot{x} = \omega(\dot{x}^2 + \dot{y}^2) = \omega V^2$$

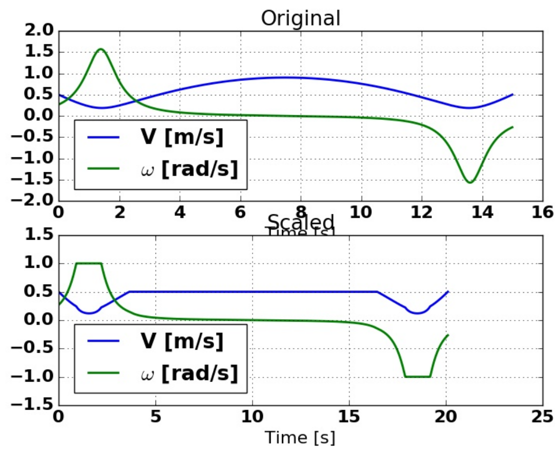
$$\omega = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{V^2}$$

Following the expressions of θ, V and ω derived above, it is trivial to implement.

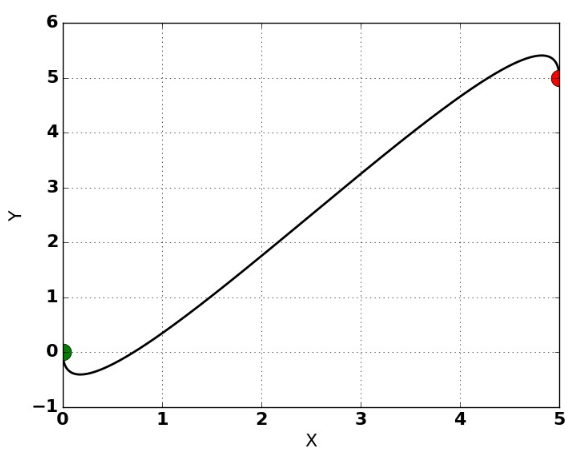
(v) The plots are shown below:



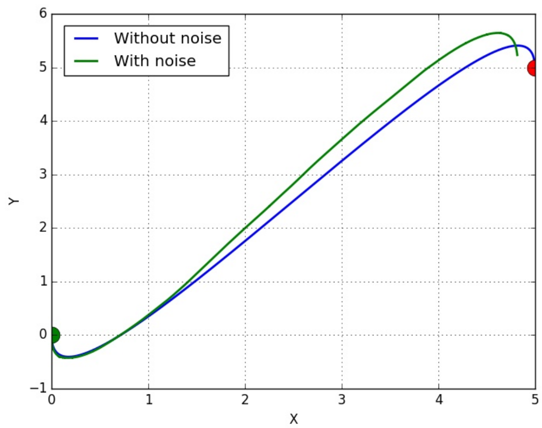
(a) arc-length history



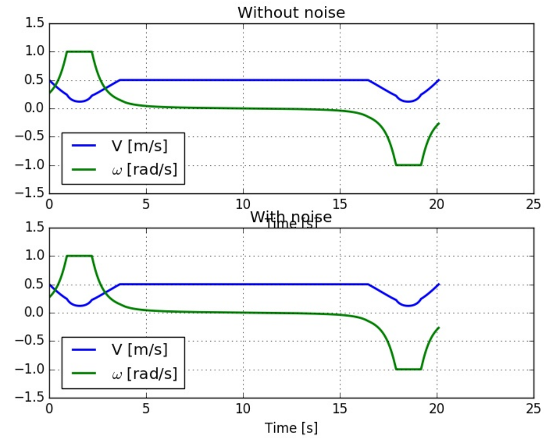
(b) control history



(c) solution trajectory



(d) simulation trajectory



(e) simulation control history

Problem 3: Closed-loop Control I

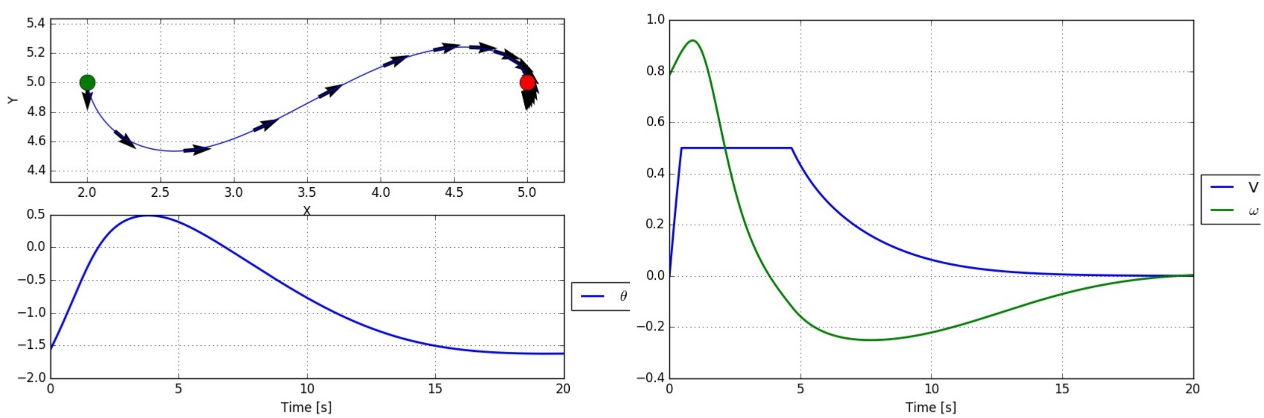
(i) See the code in `P3_pose_stabilization.py`.

(ii) For the three conditions of parking, the initial positions and poses used are

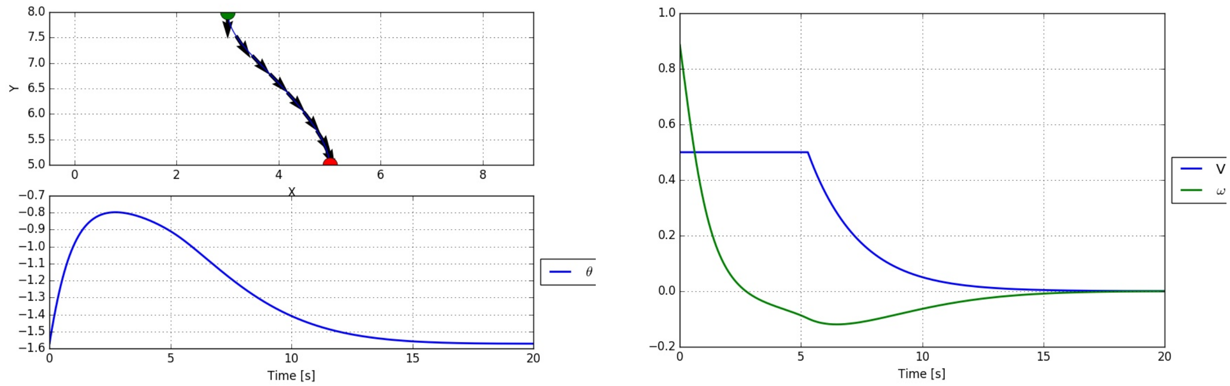
Parking Type	x_0	y_0	θ_0
Forward	3.0	8.0	-1.57
Reverse	3.0	2.0	-1.57
Parallel	2.0	5.0	-1.57

and the final time t_f is set to 20.

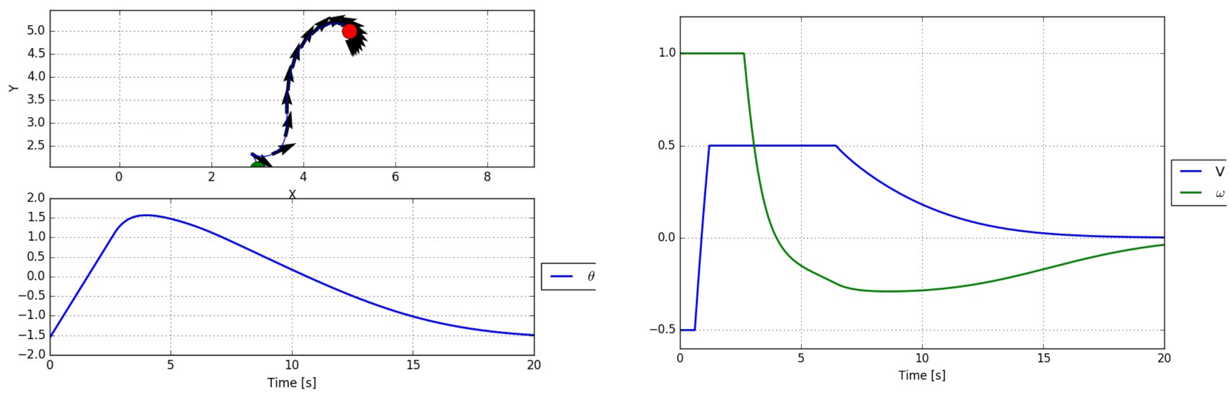
The generated plots are shown below:



(a) parallel parking



(b) forward parking



(c) reverse parking

Problem 4: Closed-loop Control II

(i) Substitute $\ddot{x}$, $\ddot{y}$ with u_1, u_2 in Eq.(1) and Eq.(2), we have

$$u_1 = a\cos\theta - \omega(V\sin\theta) \tag{3}$$

$$u_2 = a\sin\theta + \omega(V\cos\theta) \tag{4}$$

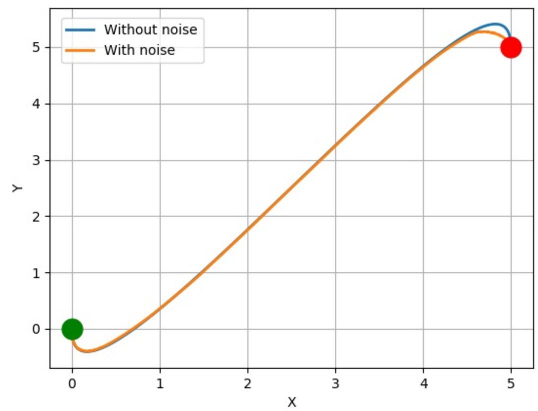
Do (3) $\times \cos\theta +$ (4) $\times \sin\theta$, we have

$$a = \dot{V} = u_1\cos\theta + u_2\sin\theta$$

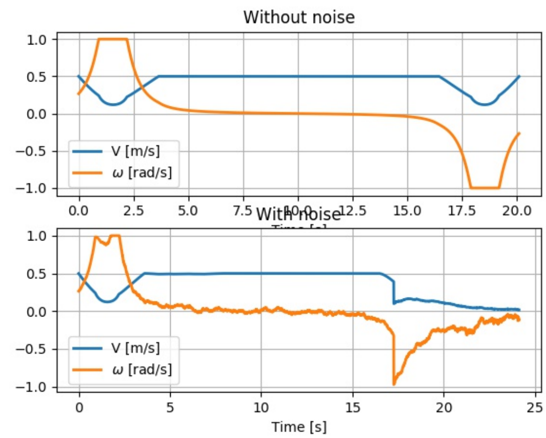
Do (4) $\times \cos\theta -$ (3) $\times \sin\theta$, we have

$$\omega = \frac{u_2\cos\theta - u_1\sin\theta}{V}$$

(iv) The plots are shown below:



(a) simulation trajectory



(b) simulation control history

Problem 5: Robot Operating System

(i) The bag file was saved as `random_strings.bag`.

(ii) The command is `rosbag play <filename>.bag`.

(iii) The bag file was saved as `turtlebot.bag`.